Exercise 1. (1) (a) Give the list of minimal $2$–cyclotomic cosets modulo $9$ which permit to classify cyclic codes of length $9$ over $\mathbb{F}_2$.

(b) How many cyclic codes (including trivial ones) of length $9$ over $\mathbb{F}_2$ does there exists?

(2) (a) Give the list of minimal $3$–cyclotomic cosets modulo $13$.

(b) How many cyclic codes (including trivial ones) of length $13$ over $\mathbb{F}_3$ does there exists?

(c) Prove the existence of a $[13, 4, \geq 7]_3$ cyclic code and a $[13, 7, \geq 5]_3$ cyclic code.

Exercise 2. A code $C \subseteq \mathbb{F}_q^n$ is said to be non degenerate, if for any $i \in \{1, \ldots, n\}$, there exists $c \in C$ such that $c_i \neq 0$.

(1) Reformulate the notion of being non degenerate in terms of a generator matrix of $C$.

(2) Reformulate the notion of being non degenerate in terms of the minimum distance of $C^\perp$. Justify why this reformulation is equivalent.

Given a non degenerate code $C \subseteq \mathbb{F}_q^n$ and a position $i \in \{1, \ldots, n\}$, the locality of $C$ at $i$ is defined as

$$\text{Loc}(C, i) := \min \{w_H(c) \mid c \in C^\perp, c_i \neq 0\} - 1,$$

where $w_H(x)$ denotes the Hamming weight of $x$. Next, the locality of $C$ is defined as

$$\text{Loc}(C) = \max_{i=1, \ldots, n} \{\text{Loc}(C, i)\}.$$

(3) Prove that $\text{Loc}(C) \geq d_{\text{min}}(C^\perp) - 1$, where $d_{\text{min}}(\cdot)$ denotes the minimum distance.

(4) Prove that $\text{Loc}(C) \leq \dim(C)$.

(5) Prove that $C$ is MDS if and only if, $\forall i \in \{1, \ldots, n\}$, $\text{Loc}(C, i) = \dim(C)$.

Given $I \subseteq \{1, \ldots, n\}$ the puncturing and shortening of a code $A$ at $I$ are defined as

$$\mathcal{P}_I(A) := \{(a_i)_{i \in \{1, \ldots, n\} \setminus I} \mid a \in A\} \quad \text{and} \quad \mathcal{S}_I(A) := \{(a_i)_{i \in \{1, \ldots, n\} \setminus I} \mid a \in A \text{ and } \forall i \in I, a_i = 0\}.$$

We admit the following statement: for any code $A \subseteq \mathbb{F}_q$, $\mathcal{S}_I(A)^\perp = \mathcal{P}_I(A^\perp)$.

(6) Let $C$ be a non degenerate code and $I \subseteq \{1, \ldots, n\}$. Prove that $\text{Loc}(\mathcal{S}_I(C)) \leq \text{Loc}(C)$.

(7) Let $c \in C^\perp$ with $c_1 \neq 0$, $w_H(c) = \text{Loc}(C, 1) + 1$ and $I \subseteq \{1, \ldots, n\}$ be the support of $c$, i.e.

$$I := \{i \mid c_i \neq 0\}$$

Prove that $\mathcal{S}_I(C)$ is an $[n - \text{Loc}(C, 1) - 1, k - \text{Loc}(C, 1)]_q$–code.
(8) Let \( t = \lceil \frac{n}{\ell} \rceil - 1 \). **Until the end of the exercise, we suppose that** \( n > (\ell + 1)t \). Prove that there exists a finite sequence of distinct indexes \( i_1, \ldots, i_t \in \{1, \ldots, n\} \) and a sequence \( c_1, \ldots, c_t \in C^\perp \) such that:

(i) for any \( j \in \{2, \ldots, t\} \), \( i_j \) is not contained in the supports of \( c_1, \ldots, c_{j-1} \);

(ii) for any \( j \in \{1, \ldots, t\} \), \( w_H(c_j) = \text{Loc}(C, j) + 1 \).

(9) Let \( s \in \{1, \ldots, t\} \) (where \( t \) has been defined in Question 8). Let \( I_s \) be the union of the supports of \( c_1, \ldots, c_s \) and \( [n_s, k_s, d_s] \) be the parameters of \( S_{I_s}(C) \). Prove that \( d_s \geq d \) and \( n_s - k_s \leq n - k - s \).

**Hint.** Use Question 7 and proceed by induction on \( s \).

(10) Let \( \ell \) be the locality of \( C \). Prove that the parameters \( [n, k, d] \) of \( C \) satisfy

\[
d \leq n - k - \left\lceil \frac{k}{\ell} \right\rceil + 2.
\]

**Hint.** Consider the shortening of \( C \) at the union of the supports of the words \( c_1, \ldots, c_t \).

Exercise 3. Let \( n \) be a positive integer, \( \sigma \) be a permutation on \( n \) elements and \( \phi_\sigma \) be the linear map:

\[
\phi_\sigma : \left\{ \begin{array}{ccc}
F_q^n & \longrightarrow & F_q^n \\
(x_1, \ldots, x_n) & \mapsto & (x_{\sigma(1)}, \ldots, x_{\sigma(n)})
\end{array} \right..
\]

(1) Show that if \( C \subseteq F_q^n \) is a code, then \( C \) and \( \phi_\sigma(C) \) have the same weight distribution.

We aim at solving the following problem:

**Problem:** *Given two codes \( C, D \), is there a permutation \( \sigma \) such that \( D = \phi_\sigma(C) \)？*

(2) Propose a naive brute force algorithm to solve the problem and compute its complexity.

(3) Prove that if two codes \( C, D \) satisfy \( D = \phi_\sigma(C) \), then,

(i) \( D^\perp = \phi_\sigma(C^\perp) \);

(ii) \( D \cap D^\perp = \phi_\sigma(C \cap C^\perp) \).

(4) Consider the following algorithm.

- if \( C \cap C^\perp \) and \( D \cap D^\perp \) do not have the same weight distribution, return false.

- else return true

(a) Does this algorithm always solve the problem?

(b) Express the complexity of this algorithm in function of the dimension \( s \) of \( C \cap C^\perp \). We suppose that the computation of the weight of a word costs \( O(n) \) and that the best manner to compute the weight distribution is to enumerate all the codewords.

(c) Explain the advantages and possible drawbacks of comparing the weight distributions of \( C \cap C^\perp \) and \( D \cap D^\perp \) instead of comparing those of \( C, D \)?

(5) Given a code \( C \) and \( i \in \{1, \ldots, n\} \), we denote by \( C_i \) the code obtained by removing the \( i \)-th entry of any codeword of \( C \). Namely:

\[
C_i = \{ (c_1, \ldots, c_i-1, c_{i+1}, \ldots, c_n) \mid (c_1, \ldots, c_n) \in C \} \subseteq F_q^{n-1}
\]

Using these codes \( C_i \) the algorithm can be refined as follows: if \( C \cap C^\perp \) and \( D \cap D^\perp \) have the same weight distribution, then compute the weight distributions of \( C_i \cap C_i^\perp \) and \( D_i \cap D_i^\perp \) for all \( i \in \{1, \ldots, n\} \).

(a) If the weight distributions of the codes \( C_i \cap C_i^\perp \) for \( i \in \{1, \ldots, n\} \) are distinct, explain why is it possible to solve the problem.

(b) If not, what kind of information on \( \sigma \) (if exists) can we get?

(c) Suppose that there exists a cyclic code \( E \) and permutations \( \sigma_1, \sigma_2 \) such that \( C = \phi_{\sigma_1}(E) \) and \( D = \phi_{\sigma_2}(E) \). Show that in this situation, the previous refinement will not be helpful.

(d) In the case of a cyclic code as described in Question (5c), propose an improvement of the refinement which may solve the problem.